Achieving Peer-to-peer Telecommunication Services through Social Hashing

Xiaohui Yang[1], Ram Dantu[2], Duminda Wijesekera[1]
[1] Department of Computer Science
George Mason University, Fairfax, VA 22030, USA
[2] Department of Computer Science and Engineering
University of North Texas, Denton, TX 76203, USA
xyang3@gmu.edu, rdantu@unt.edu, dwijesek@gmu.edu

***Abstract*—Although peer-to-peer (P2P) Internet Telephony gains more and more market share, supporting traditional telephony related use cases is an indispensable requirement. However, communication services designed for this purpose are usually traditional circuit-based, and the centralized structure makes it nearly impossible for their deployments in a distributed, unsecure telecommunication environment.**

This paper proposes an approach for achieving various kinds of communication services on P2P voice over IP (VoIP) systems by building trust and executing supervision through social networks. We present a system architecture design for network topology maintenance and security assurance. Social protocols and social computing are used to study how P2P entities can be efficiently mapped to social networks, and how social functionalities can benefit communication services implementation. To demonstrate the approach feasibility, we exemplify this architecture in P2P VoIP emergency services by using gossiping and membership management techniques. We believe that our proposed approach will be able to support diverse services on P2P VoIP systems with the performance and security guaranteed to compete with centralized telecommunication systems.

I. INTRODUCTION

Telecommunication services such as Spam control, Emergency service, Lawful Interception, billing, voicemail, etc, are of vital importance to the telephone functionality. But due to P2P VoIP's distributed nature, the lack of identities assurance and certain centralized control is the biggest obstacle for the telecommunication services implementation.

We propose to use social hashing: a collaborative approach for building trust and executing supervision in P2P DHT networks. Social hashing can help to solve the problem by exploiting clustering techniques and hub-like structure in social networks. In addition, incorporating social ties into DHT can make for efficient routing and balance the trade-off between security and performance.

The main contribution of this paper is to present a novel concept of correlating telephone network with social network based on the structural properties. It is the first to use social functionalities to solve the difficulties of implementing telecommunication services. We aim at achieving reliable and efficient telephone communication services through the perfect union of social and computer networks.

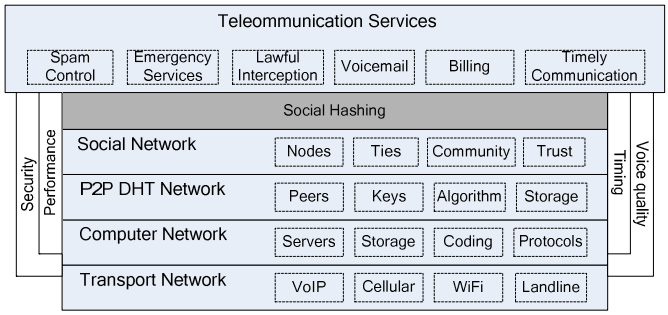

Fig. 1. Architecture for the socialized P2P VoIP communication services

II. DESIGN

A. Proposed Architecture

Figure 1 depicts the architecture design for the peer-to-peer telecommunication services implementation. From bottom up, this architecture shows a layer-by-layer logical abstraction and the evolution of telephone communication as well. The nethermost two layers provide the transport and network layers as the necessities for telecommunication. As a particular kind of computer networks, P2P DHT layer consists of peers that maintain the connectivity and process communication through DHT-based routing. We propose creating a social network among these peers for the next layer consisting of telephone contacts. Relationships that evolve from social roles provide a reliable and efficient platform for the upper P2P telecommunication services implementation.

B. Social Methodology

***Structural properties of call graph*:** based on the assumption of steady user behavior, P2P VoIP user call graphs can be well modeled by small world networks [1]. We define the degree d_i as the number of contacts of node i. The degree distribution follows a power-law distribution, which induces a clusterizing structure for the whole telephone social networks. Link degree correlations can be calculated from the scale-free metric $s \in [0, 1]$ [2] and the Pearson correlation coefficient $r \in [-1, 1]$ [3]. These properties show a structure of a hub-like "core" of the inter-connected high-degree nodes, with the lower-degree nodes scattered on the network fringe [4]. This core-edge structure is the basis for our social hashing, which presents a collaborative centralized control on distributed systems.

Social DHT: the application of social hashing is mainly reflected in the security enforcement and routing services. We

propose to use society based Node ID assignment and introduce the contacts oriented social tables for the <*key, value*> pair lookup. Our social DHT is based on Bamboo DHT [5], and we use social call graph to intelligently assign a numeric identifier $[1, 2^{160})$ to each social member. Treating each identifier as a sequence of digits of base 2^b, and the ID space is divided based on the social cluster. Members of the same cluster share the same prefix digits, and the hub node is assigned the lowest identifier in each cluster. Nodes on DHT overlay can be correlated based on their social tables, which list users' contacts and their corresponding degrees (the number of contacts).

Social routing: to route a message with key *k*: 1) the node checks *k*'s existence in regular Bamboo leaf set and its own social table; 2) if not found, checks whether *k* falls in its own cluster; 3) if yes, follows intra-cluster routing by forwarding message along with a series of highest degree contacts; 4) otherwise, forwards the message according to the hub node based inter-cluster routing (shown in Fig. 2); 5) if not found, use Bamboo routing table for message forwarding.

Based on it, social distance SD_{ij} from node *i* to *j* can be calculated as (1):

$$SD_{ij} = \begin{cases} \mu * \sum(1 - d_{k_m}/d_{h_1}), \ i,j,k_m \in C_1 \\ (SD_{ih_1})_{Min} + SD_{h_1h_2} + (SD_{h_2j})_{Min}, \ i \in C_1, j \in C_2 \end{cases} \quad (1)$$

Where μ is a constant unit distance factor, d_{k_m} is the degree of each intermediate node $k_{1..n}$ on routing path, and h_1, h_2 is the hub node of cluster C_1, C_2 respectively.

Meanwhile, message should be routed efficiently with little overhead and the minimum hop count. We calculate the routing distance D_{ij} from node *i* to node *j* according to (2):

$$D_{ij} = Min\{SD_{ij}, R_{ij}, (SD_{ik} + R_{kj})_{Min}\} \quad (2)$$

where R_{ij} is the routing table based routing distance, and the choose of *k* is the guarantee for *O(logn)* efficient routing.

***Trust*:** we assume that trust exists between two adjacent friends. Hub nodes in each cluster can be highly trusted as an authority, whereas trust from authority *a* to fringe node *i* will heavily rely on their social distance SD_{ai}. Intuitively, trust degree $t_{ij} \in [0, 1]$ from node *i* to node *j* is proportional to $\lambda e^{-\lambda*(SD_{ij})_{Min}}$, where λ is a constant referred as the trust scale, and $(SD_{ij})_{Min}$ is the minimum social distance between node *i* and *j*. Nodes may dynamically update the trust to other nodes based on the call behavior e.g. call frequency, etc. Let $w \in [0, 1]$ be a measure of the latest t_{ij}, the node trust degree *T* may be updated by using the exponential moving average (3):

$$T_n = T_{n-1} + \alpha * (w - T_{n-1}) \quad (3)$$

where $n \geq 2$ shows the time period for the observation, and α is the degree of trusting decrease.

Service supporting: hub-like structure introduces a virtual core into a distributed P2P VoIP network, which can be used to solve the difficulties in implementing telecommunication services. First, the trustworthiness of each node is the basis for successful *Spam* control and secure *Voicemail* storage. Second, some "choke points" may be set at the hub nodes for attack control, *Lawful Interception*, and *Billing* service. Third, hub nodes with the highest degree can be elected as the leaders for message dissemination and the answering points for *Emergency Services*.

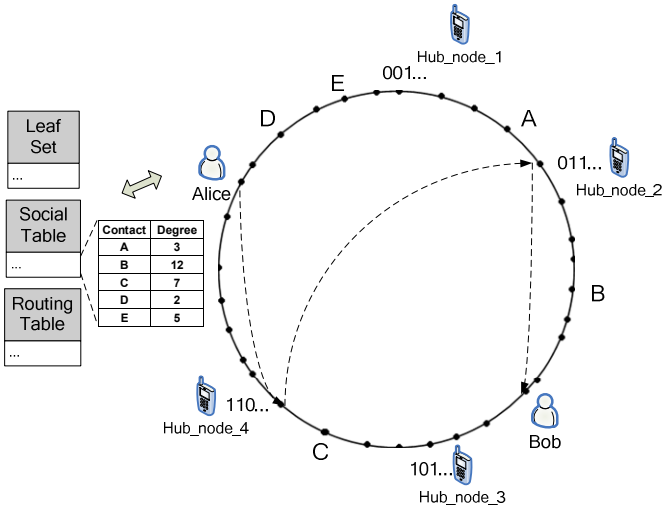

Fig. 2. Social table based routing on social DHT from Alice to Bob

III. Future Case Study

The proposed case study aims at proving the validity of our social hashing approach. We hope to discover the perfect union of social and computer networks to benefit telephone communications. Reverse 911, an emergency notification service, will be implemented on social DHT in our study. We will use social hashing to solve the primary difficulty of ascertaining users' physical location in real-time.

To be applied for Reverse 911, users on social network are grouped based on their physical location. Hub nodes that have a high probability of location-distributed can be used as an authority for information dissemination. User mobility is supported by subscribe/unsubscribe to the location group. Users in one group will be highly trusted by other members. Group membership will be checked constantly based on the service boundary, and only members in certain area can be served. We propose to use Gossip protocol [6], for information dissemination and membership management. We believe that our social gossiping can help to achieve the notification with the tradeoff of security vs. performance.

Acknowledgment

This work is supported by the National Science Foundation under grants CNS-0627754, CNS-0619871and CNS-0551694. Any opinions, findings, conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation.

References

[1] D.J. Watts, Collective dynamics of 'small-world' networks. Nature, 1998.
[2] L. Li, Towards a theory of scale-free graphs: definitions, properties, and implications. Internet Mathematics, 2006, pp. 2(4): 431–523
[3] M.E.J. Newman, Assortative mixing in networks. Physical Review Letters 89, 2002, 208701.
[4] A.A. Nanavati, On the structural properties of massive telecom call graphs: findings and implications. ACM CIKM, 2006.
[5] S. Rhea, Handling churn in a DHT. Proceedings of the USENIX Annual Technical Conference, 2004.
[6] R.Th. Eugster, Lightweight probabilistic broadcast. ACM Transactions on Computer Systems, 2003, pp. 341–374.